

CD-MAKE 2017

# The more the merrier - Federated learning from local sphere recommendations

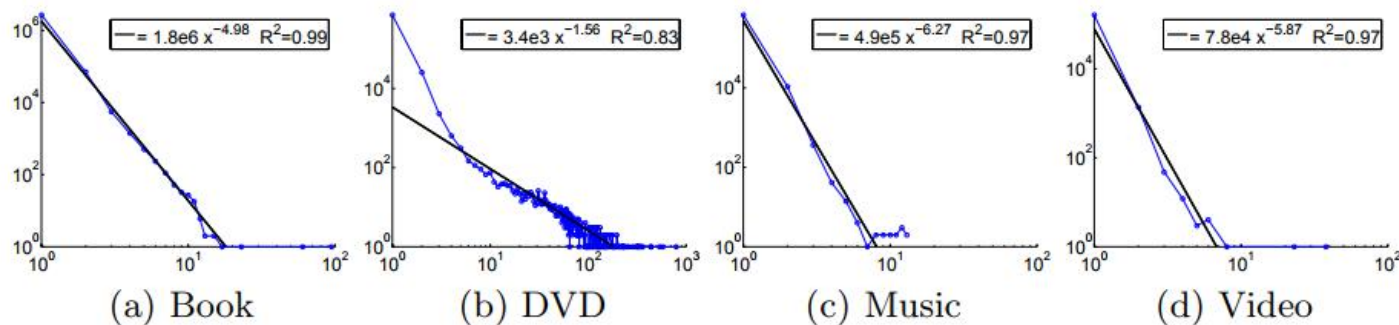


Bernd Malle, Nicola Giuliani, Andreas Holzinger

[b.malle@hci-kdd.org](mailto:b.malle@hci-kdd.org)

- Thinking about Machine learning from a \*European\* startup perspective
- Which brings a few particular challenges with it:
  - Usually less startup capital than U.S. competitors  
=> therefore less money for computing power
  - much fewer possible customers (initially) than Asian competitors
  - GDPR is a major impediment => expressly prohibits use of personal data...
- Maybe we can circumnavigate all those hurdles via  
**Client-side Machine Learning**

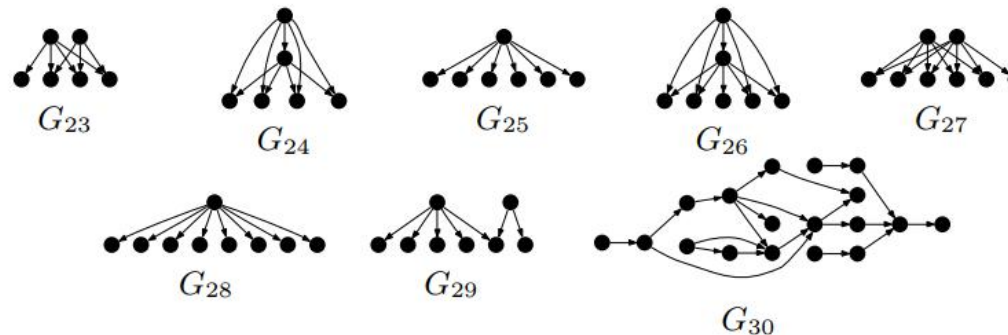
- How far away are relevant decision points within a social networks usually?
- Leskovec, even back in 2006 observed recommender cascades within an online shopping system
  - for 3 out of 4 products: maximum size of cascade was  $< 10$



**Fig. 1.** Size distribution of the cascades for the four product types (log size of cascade vs. log count). Superimposed line presents a power-fit.  $R^2$  is the coefficient of determination.

Leskovec, Jure, Ajit Singh, and Jon Kleinberg. "Patterns of influence in a recommendation network." Pacific-Asia Conference on Knowledge Discovery and Data Mining. Springer, Berlin, Heidelberg, 2006.

- most were not chains, but one node influencing many others (splits) or several recommendations directed at one node (merges)
- single recommendations made up the majority of 'cascades'
- overall, the average ego network from which relevant recommendations originated was little more than 1(!)

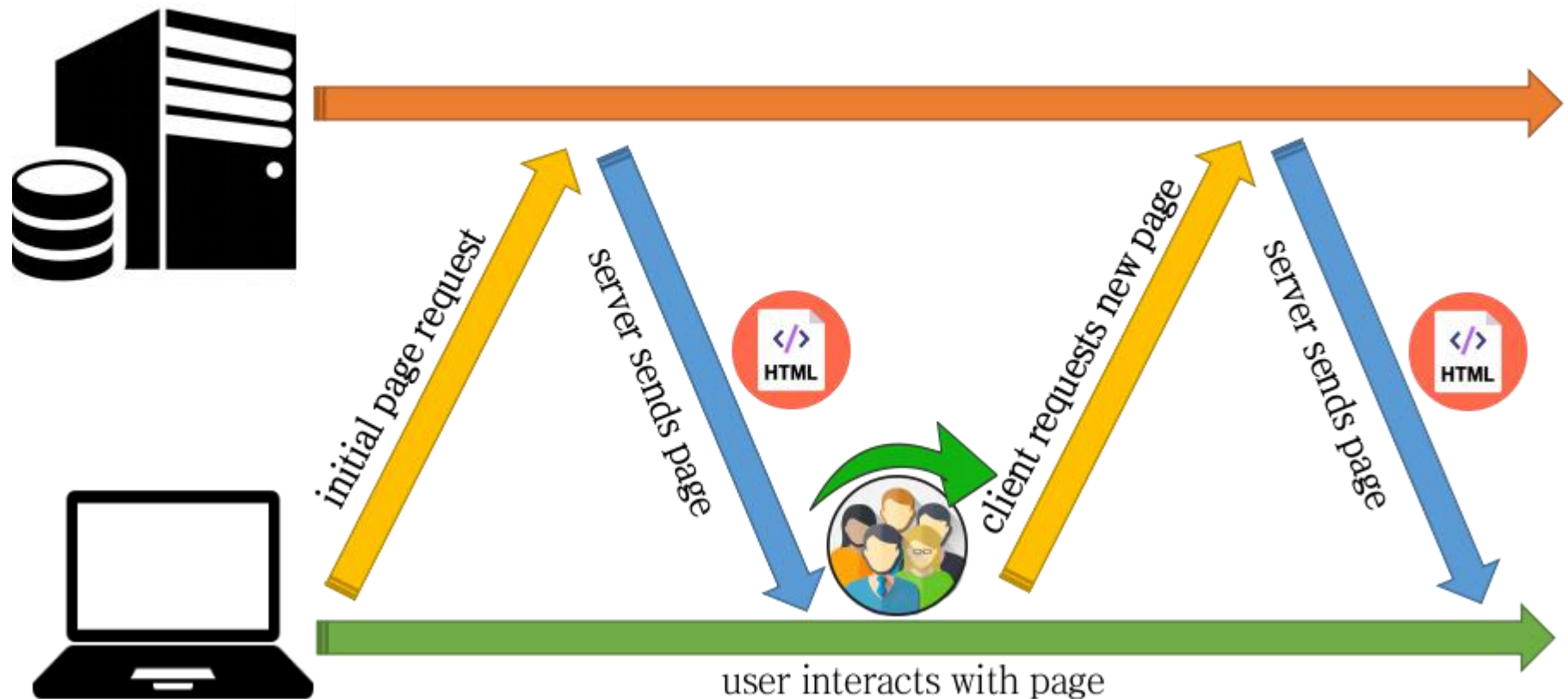


**Fig. 2.** Typical classes of cascades.  $G_{23}$ ,  $G_{27}$ : nodes recommending to the same set of people, but not each other.  $G_{24}$ ,  $G_{26}$ : one node recommends to another, and both recommend to the same community.  $G_{25}$ ,  $G_{28}$ ,  $G_{29}$ : a flat cascade.  $G_{30}$  is an example of a large cascade.

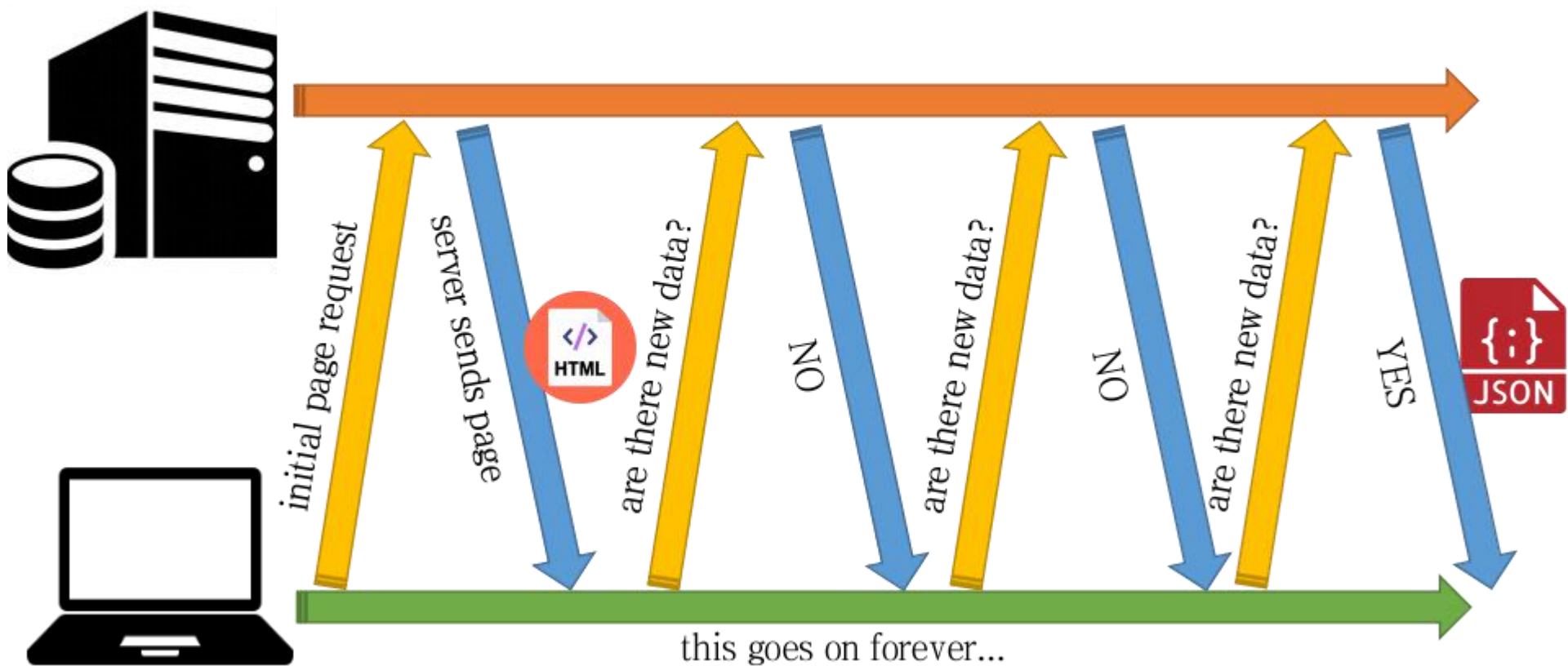
Leskovec, Jure, Ajit Singh, and Jon Kleinberg. "Patterns of influence in a recommendation network." Pacific-Asia Conference on Knowledge Discovery and Data Mining. Springer, Berlin, Heidelberg, 2006.

- the FB graph has been estimated to have a diameter of as low as 4
- because of many loose contacts instead of friends
- the diameter is shrinking with new connections
- although the graph is globally very sparse, individual node neighborhoods contain surprisingly dense structure
- Conclusion: We don't need the whole graph to calculate good recommendations - it might be possible to just take a node's immediate vicinity into account!

## Traditional request / response model < 2005



## Longpolling via Ajax 2005 - 2012 (& still in use)





# Modern Pub/Sub with constant synchronization

```
class GoalsCollection extends Mongo.Collection {
  insert(goal, callback) {
    return super.insert(goal, callback);
  }
}

export const Goals = new GoalsCollection('goals');

Goals.schema = new SimpleSchema({
  _id: { type: String, regex: SimpleSchema.RegEx.Id },
  name: { type: String },
  phasesCount: { type: Number, defaultValue: 0 },
  actionCount: { type: Number, defaultValue: 0 },
  userId: { type: String, regex: SimpleSchema.RegEx.Id },
});

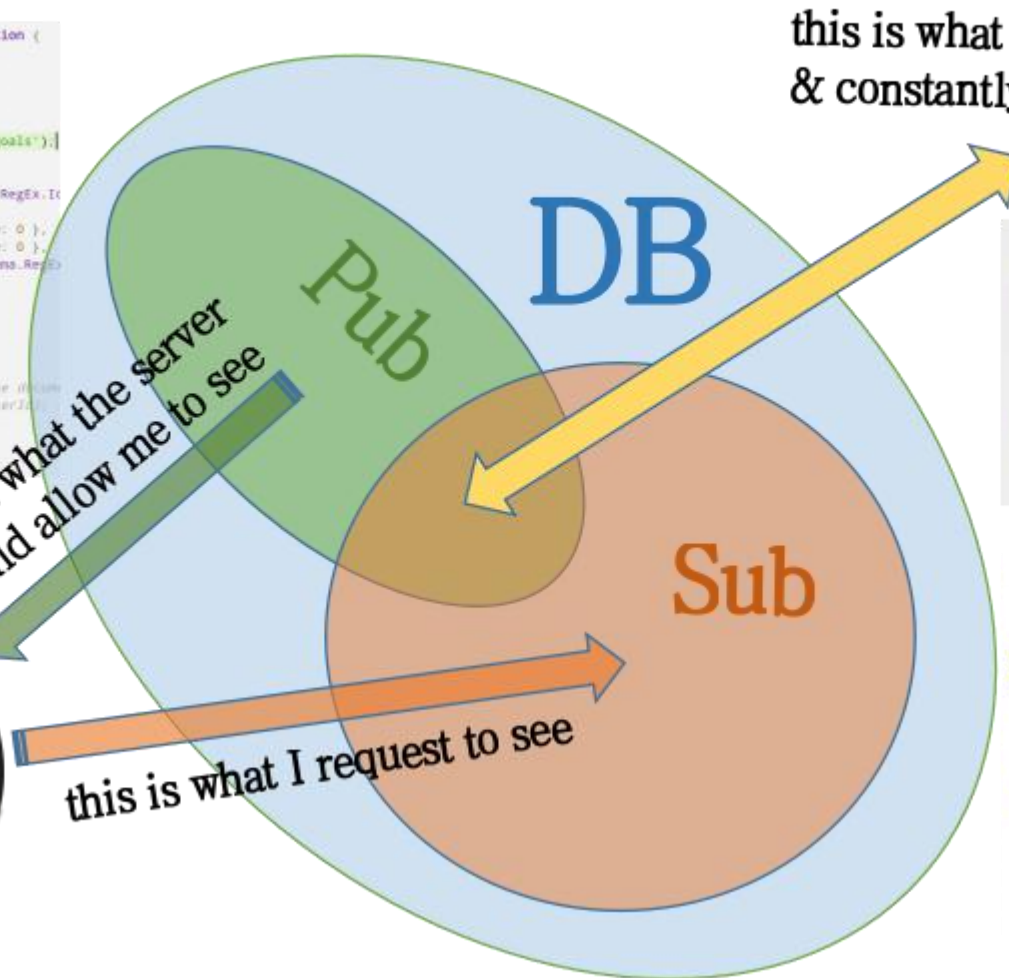
Goals.attachSchema(Goals.schema);

Goals.allow({
  insert: function (userId, doc) {
    // the user must be logged in, and the doc owner == user1
    // return (userId == doc.owner == user1);
    return true;
  }
});
```

this is what the server  
would allow me to see

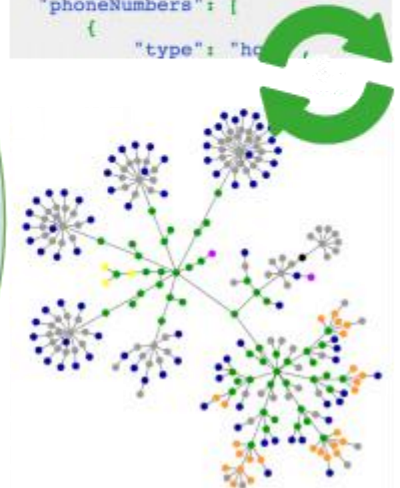


this is what I request to see



this is what the server resolves  
& constantly syncs to the client

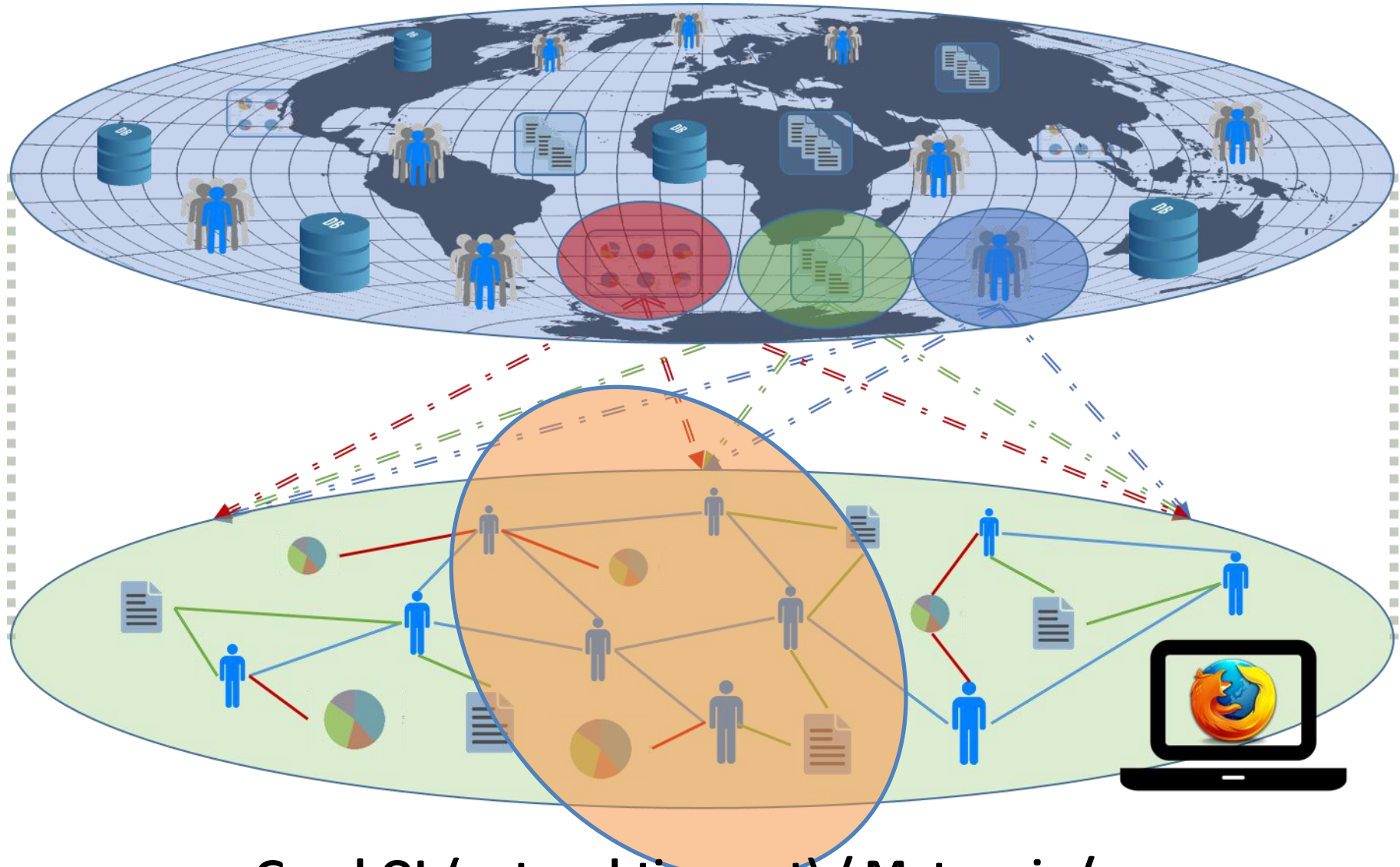
```
{
  "firstName": "John",
  "lastName": "Smith",
  "age": 25,
  "address": {
    "streetAddress": "21 2nd St",
    "city": "New York",
    "state": "NY",
    "postalCode": "10021"
  },
  "phoneNumbers": [
    {
      "type": "home",
      "number": "212 555-1234"
    },
    {
      "type": "mobile",
      "number": "917 555-1234"
    }
  ]
}
```





- This means in effect, that all information within the neighborhood of a node (if you see it graph theoretically) is constantly available within the browser / mobile device
  - my direct friends on a social network
  - all the information within my project group
- Combining those two views, we see that a majority of relevant recommendations could also be computed client-side...

- Many globally distributed databases *dont have to / cannot be* implemented as a graph (AFAIK facebook also does not store one central graph)
  - would need too many globally propagating updates  
"the consequences of a tiny little update could affect the farthest reaches of the global graph"
  - huge problem for graph databases
- The local sphere can get it's information from many pub/sub mechanism targeting different endpoints in the global sphere
- The local sphere is a superset of the actual user's data, but user's data plus it's relevant vicinity



**GraphQL (not real-time yet) / Meteor.js / ....**

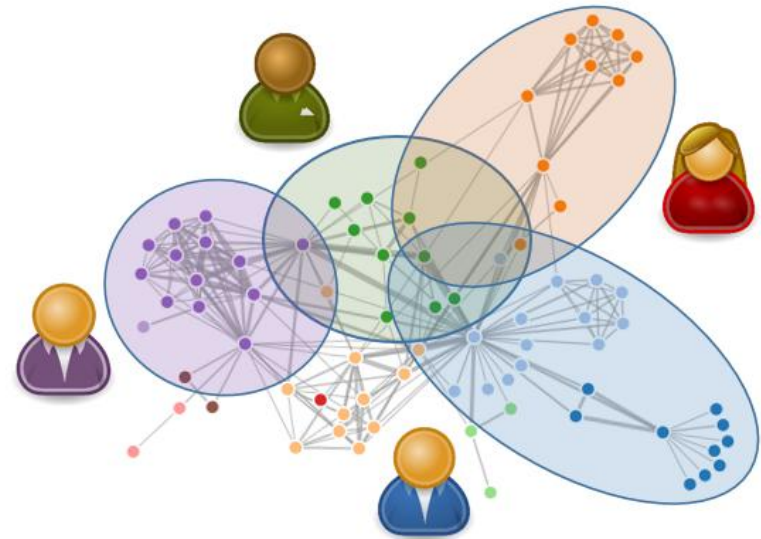
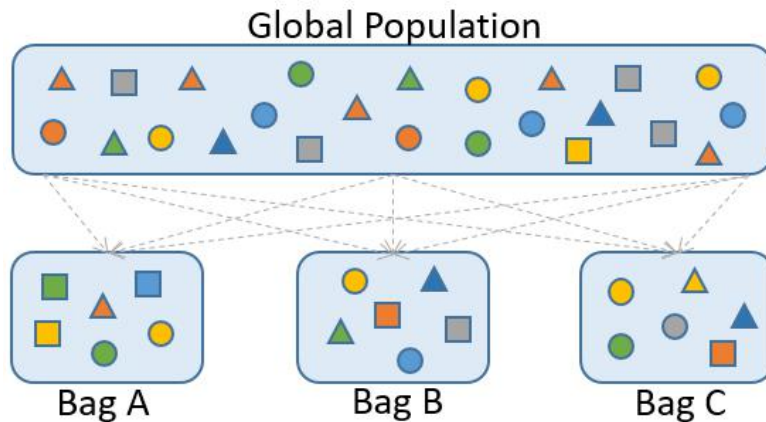
- GDPR says processing of personal data is expressly prohibited
  - but they are talking about data you **collected**
  - what if you haven't collected it because it never left the users' device?
- You can potentially use a wealth of information available on the client device you could never access server-side
  - address book
  - calendar
  - GPS
  - local files
  - .....

- BYOCP - "Bring your own computational power"
  - World's fastest supercomputer can do 93 Petaflops
  - A Geforce 1080 runs at about 12 Teraflops (single prec.)
  - You need ~7,750 customers with such cards to reach the TaihuLight
  - iPhone 7: GPU operates at 729.6 GFLOPS
  - ~130k iphones stack up to the TaihuLight
- of course those are very superficial & unrealistic numbers, but they give a good feeling about the order of magnitude we're talking about
- a few hundred thousand users is not much for a successful startup today => SCALABILITY !!!

1. Pub/Sub keeps the local sphere in sync with global data
2. Local algorithms compute recommendations
  - these can also come from a client-side crawler (like the FB crawler which scans URLs you paste into a comment field and extracts images from a website etc.)
1. Upon user acceptance, a new node is introduced into the local sphere + updated to the global sphere
2. Client devices with overlapping local spheres now receive that node in the background
3. Their recommenders respond...



- Maybe it's even possible to implement Machine Learning paradigms on such a distributed platform (see bagging below)
- In the end, we might not even have to curate a global graph anymore - it could be a 'ghost-like', implicit instantiation of the sum of all local spheres....





# Thank you!